Analysis of Amazon Co-Purchases

A Detailed Look at the Amazon Product Statistics and a Neural Network of Amazon Co-Purchasing Algorithms

Colton Morley, Jason Brown
Department of Computer Science and Engineering, University of Nevada, Reno, NV

Abstract – This project will analyze the data spread of Amazon's products, visualize their relationships, and attempt to create a neural network that can analyze which products are likely to be purchased together and which product information is likely to be true, given other attributes. It will also take into account the reviews and ratings to see how the algorithms decide which product to recommend when comparing similar products.

I. Introduction

Amazon.com has amassed the largest customer base of any company in history and sells millions of products per minute. Aside from numerous significant business strategies which made the company effective, their product recommendation algorithms have undoubtedly proven their success in making customers spend more money on items they feel that they need. Using machine learning and classifying algorithms, Amazon is able to track a customer's purchases and learn what they might want to buy next. This is part of a sizable neural network which takes into account every customer's shopping habits. Amazon will recommend products that both relate your previous purchasing habits and millions of other customer's similar purchases, learning from each click as you check out. The large-scale success which has been achieved with the help of these algorithms is what makes a product recommendation algorithm so significant and valuable. The theory applied in this product recommendation algorithm can be applied in most instances of engineering, business, and predictive modeling. Whether it is product recommendation, search results, media viewing or fraud detection, the algorithms used in these operations are universally valuable. Team *R10* will analyze the methodology of these algorithms and data which they pertain to in Amazon's product meta-data.

To begin breaking down the data, it is important to know more about the dataset at hand. To analyze this, Team R10 created a subnet of nodes, where each node represents a product and each connecting line represents the co-purchasing relationship between two products. After creating this subnet, the team conducted numerous tests to learn about how the product data is relatable. These tests included node degree, node closeness centrality, node betweenness centrality, and node eigen centrality. These algorithms broadly describe the influence that any particular node has on the rest of its network. Team R10 began with researching which data would best fit the goal. We were given a dataset with data such as product sales rank, review count, product rating, and co-purchased items.

II. Data Description and Key Observations

The data that Team R10 used to evaluate this topic was from Stanford's Network Analysis

Project. SNAP is a small database consisting of work from both undergraduate and graduate students at Stanford University which all pertain to neural networks and related data[1]. The dataset we used was a text file of all product data and their according relationships; product data consisted of Amazon category ID, Amazon identification number, category, product title, salesrank, co-purchased items, category and product specific information, review count, and review rating. And all of these product attributes were provided for only the following product categories: music CDs, DVDs, books, and VHS video tapes. Although the original metadata provided by SNAP was in the form of a text file, the team was able to find the exact same data in CSV format, proving to be much more organized and effective for data traversal and evaluation. This CSV was even further simplified into a dataframe from the Panda's library to quickly divide, evaluate, and traverse.

After this data is stored as an approachable dataframe, it is analyzed to determine the spread of the data. Multiple algorithms like node degree, node closeness centrality, node betweenness centrality, and node eigen centrality were applied to the data to understand the relational influence that any one node might have on the rest of the network. The results were that the product data was evenly spread, proving that any few nodes don't have the power and influence to dramatically affect the rest of the network analysis algorithms.

Below are the visualization plots for each of the data visualizations:

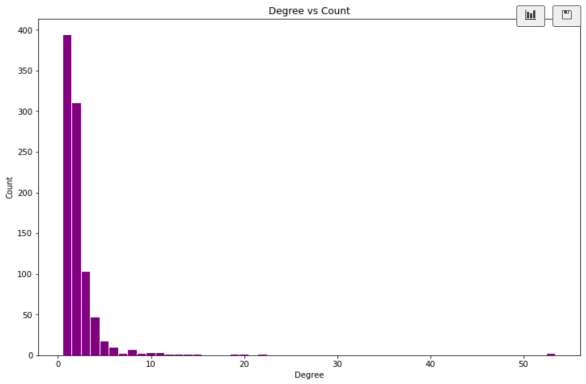

Fig.1: Node Degree

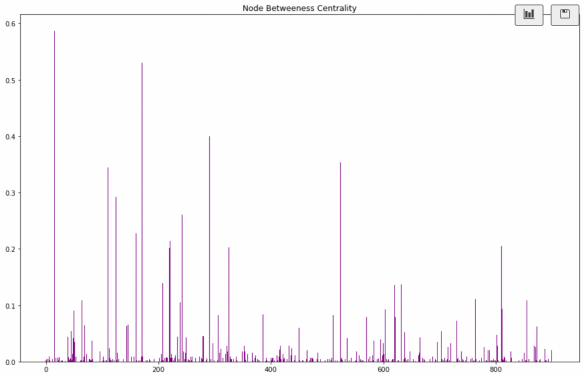

Fig 2: Node Betweenness Centrality

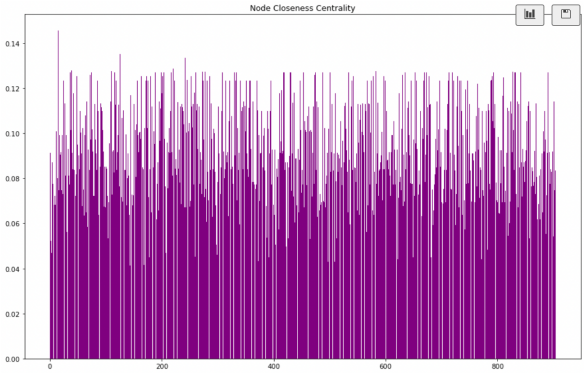

Fig 3: Node Closeness Centrality

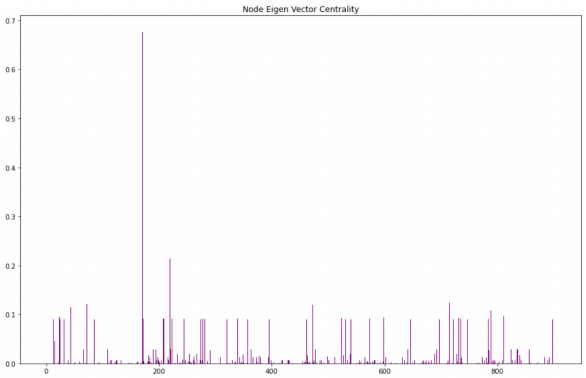

Fig 4: Node Eigen-Vector Centrality

General Plot Definitions:

- Node Degree: Number of ties to other nodes.
- Node Betweenness Centrality: A measure of centrality based on shortest paths.
- Node Closeness Centrality: Measure of path length to understand influence.
- Node Eigenvector Centrality: Measure of node influence on network.

III. Correlation Between Data Variables

The variables used in this analysis are salesrank, number of reviews, number of downloads and average rating. Correlation analysis between all the variables found that the review count and number of downloads were the most strongly correlated variables in the dataset. After this pair is Amazon salesrank and rating which are negatively correlated. The correlation between all variables in the analysis is shown below:

Table 1. Correlation between variables

	Salesrank	Reviews	Downloads	Rating
Salesrank	1.000	-0.098	-0.101	-0.395
Reviews	-0.098	1.000	0.9524	0.098
Downloads	-0.101	0.9524	1.000	0.101
Rating	-0.395	0.098	0.101	1.000

When splitting the data was divided by product categories, the correlations remained consistent throughout the different groups. The correlation between reviews and downloads remained the strongest relationship between any of the product attributes. It can also be noted that salesrank has a negative correlation with its related attributes because a smaller salesrank number is better than a larger salesrank number.

IV. Agglomerative Clustering Algorithm

The first clustering algorithm applied to the dataset was Agglomerative Clustering. This algorithm starts by separating each data object as a single cluster, then merging similar clusters until all objects fall within one grouping [2]. This is also called a 'bottom-up' approach. To do this, Team R10 used a KNeighbors Graph as a connectivity matrix for the classifier.

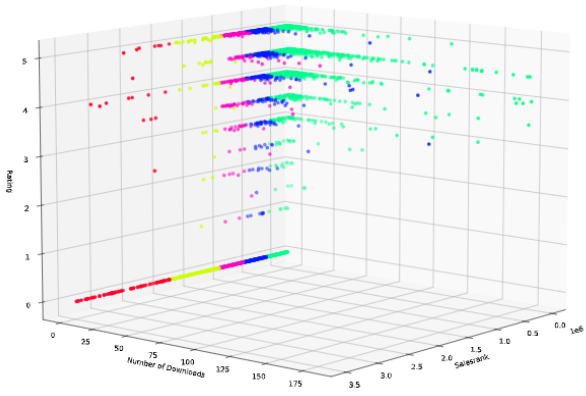

Fig 5. Agglomerative Clustering Applied to All Product Categories (2000 data points)

The above visualization of Agglomerative Clustering clearly shows that sales rank increases along with the increase of a product's rating. Inversely, it can be seen that downloads do not have a significant effect on rating. This provides a direct correlation between these two variables and their one-way relationship.

V. K Means Classifying Algorithm

The next classifying algorithm that was applied to the Amazon Co-Purchasing Data was the K-Means Clustering Algorithm. This unsupervised

algorithm immediately clusters data objects based on similar features [2].

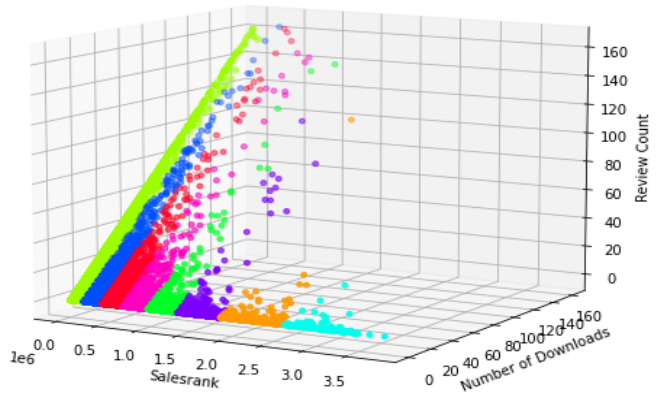

Fig 6. K Means Classifier Applied to All Product Categories (25,000 data points)

As seen in the above plot representing the results of the K-Means Classifier, it is blatant to see that rating performance and the number of downloads have a direct and an extremely strong relationship. This can be shown as the slope between the two axes is about 1:1 as the plot depicts a clear uptrend. It can also be observed that a greater sales rank leads to more ratings of higher quality.

VI . Neural Network to Predict Salesrank

The next analysis conducted on the data was the creation of a multilayer perceptron (MLP) regressor. MLP is a class of Artificial Neural Network (ANN) [3]. This regressor was trained on the relevant data variables, including number of downloads, rating, and review count. The regressor uses Limited-Memory BFGS solver, a hidden layer size of 6, and a regularization term of 1e-5. The results of this classifier when using a training size of 80% of the data, and a test size of 20% are listed below.

Table 2. MLP ANN Regressor Error Results

Training Size	80,000
Mean Absolute Error	0.0988
Root Mean Squared Error	0.0204

This table concludes that this Multilayer Perceptron can use given variables on a product to predict the product's salesrank with a relatively high degree of accuracy. These training and testing processes were run on a random subset of data consisting of 100,000 items due to limited machine resources. These error scores remained consistent over multiple iterations of random data selection and testing, with MAE ranging between 0.080 and 0.105, and RMSE rangining between 0.020 and 0.031.

Acknowledgment

The authors would like to acknowledge the Stanford Network Analysis Project (SNAP) for providing the data used in this analysis. They would also like to acknowledge Feng Yan and Lei Yang of the University of Nevada, Reno, who's lectures on machine learning techniques and clustering algorithms were crucial in this analysis.

References

[1] J. Leskovec, L. Adamic and B. Adamic. The Dynamics of Viral Marketing. ACM Transactions on the Web (ACM TWEB), 1(1), 2007.

[2] Lars Buitinck (ILPS), Gilles Louppe, Mathieu Blondel, Fabian Pedregosa (INRIA Saclay - Ile de France), Andreas Mueller, Olivier Grisel,

Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort (INRIA Saclay - Ile de France, LTCI),

[3] Jaques Grobler (INRIA Saclay - Ile de France), Robert Layton, Jake Vanderplas, Arnaud Joly, Brian Holt, Gaël Varoquaux (INRIA Saclay - Ile de France)

[4] P. Basuchowdhuri, M. K. Shekhawat and S. K. Saha, "Analysis of Product Purchase Patterns in a Co-Purchase Network," *2014 Fourth International Conference of Emerging Applications of Information Technology*, 2014, pp. 355-360, doi: 10.1109/EAIT.2014.11.

[5] Dellarocas, C., Awad, N., Zhang, X.M.: Using online reviews as a proxy of word-of-mouth for motion picture revenue forecasting. SSRN Electron. J. (2004)

[6] Dellarocas, C., Awad, N., Zhang, X.M.: Using online ratings as a proxy of word-of-mouth in motion picture revenue forecasting. Working Paper (2005)

[7] Forman, C., Ghose, A., Wiesenfeld, B.: Examining the relationship between reviews and sales: the role of reviewer identity disclosure in electronic markets. Inf. Syst. Res. 19(3), 291–313 (2008)

[8] Clemons, E., Gao, G, and Hitt, L. 2006. "When Online Reviews Meet Hyperdifferentiation: A Study of the Craft Beer Industry," Journal of Management Information Systems (23:2), pp. 149-171.